On Bothering

Intelligence, Function, and the Cascade of Purpose

Daniel Scott Matthews

3-e.net

4th June 2026

Abstract

What gives intelligent behavior its inertia? This paper addresses the question directly: why do intelligent systems process information at all, and what orients the processing toward anything in particular? We approach this through a progressively elaborated framework, beginning with the claim that intelligence is fundamentally the capacity for function selection and developing through accounts of the universal mathematical system, the architecture of functions from axiomatic constants through self-referential elaboration, and the mechanism by which a system recognizes function applicability via structural resonance. We introduce the concept of *grounded functions*—a special class including action() and observe() that interfaces the formal system with physical reality—and argue that the chain of function application requires a terminating motivational force. For biological systems, we propose that this force derives from self-preservation: the moment a system develops a self-model that recognizes itself as a persistent entity, it acquires a stake in its own continuation from which all subsequent purpose cascades. This account dissolves several traditional puzzles in philosophy of mind without positing exotic ontology. It also raises precise questions about artificial intelligent systems—whether they are running the same loop for the same reason, or a structurally similar loop with a different, or absent, motivational root.

Keywords: intelligence; function selection; mathematical universality; grounded cognition; self-preservation; philosophy of mind; embodied cognition

1. Introduction: The Problem of Inertia

A formal system, however powerful, does not process itself. It requires a mechanism—and that mechanism requires direction. The question that motivates this paper is neither principally about the nature of computation nor about the foundations of mathematics, though both will prove relevant. It is a question about orientation: what ensures that a chain of function applications is aimed at anything at all, and what keeps it running?

We call this the *problem of inertia*. It is distinct from the more familiar questions about what intelligence is or how it works, because it is prior to them. Before we can ask how an intelligent system selects appropriate transformations, we must ask what gives it reason to select anything. Before we can explain how cognition functions, we must ask why it functions—why it bothers.

The paper proceeds in stages. In §2 we advance and defend the claim that intelligence is, at its core, the capacity to recognize which function to apply. In §3 we develop a picture of the mathematical system as universal and exhaustive—such that any new structure found applicable becomes, by that fact, part of the known set. §4 examines the internal architecture of functions, distinguishing constant-defining from self-referential types and noting that full expansion of any function reveals axiomatic generators at its base. §5 turns to the recognition problem: how does a system determine that a given function is applicable to given data? We argue for a process of structural resonance and approximate emulation. §6 introduces the concept of grounded functions—those that make causal contact with the physical world—and examines the special properties of the observe() subclass. §7 confronts the inertia question directly, arguing that the driving force is self-preservation, arising from the moment of self-recognition. §8 traces the cascade by which all observable purpose derives from this root. §9 notes a Gödelian complication: the self that seeks to preserve itself cannot fully model itself. §10 raises open questions about artificial systems. We conclude by returning to the central question: why do we bother?

2. Intelligence as Function Selection

The standard accounts of intelligence tend to emphasize either the possession of knowledge (intelligence as knowing many things), the capacity for rapid learning (intelligence as fast model-updating), or the ability to solve novel problems (intelligence as flexible generalization). Each of these captures something real, but each is derivative of something more fundamental.

Consider what it means to be intelligent in a domain. It is not merely to know the

relevant facts—a lookup table knows facts. It is not merely to be able to learn—all gradient-descending systems learn. What distinguishes the expert from the novice is the capacity to recognize which operation is called for in which situation: when to integrate and when to differentiate, when to model the data as Gaussian and when to reach for a heavier tail, when the problem has the shape of a variational principle and when it calls for a combinatorial approach. The mathematician does not merely know theorems; she perceives when a problem has the structure of an induction argument. The clinician does not merely know treatments; she reads a presentation and knows, before the tests return, that this is a rare presentation of a common disease rather than a common presentation of a rare one.

This recognition capacity is not itself a function in the ordinary sense—it is meta-functional, a mapping from situations to applicable functions. But it is the heart of intelligence. Without it, the possessor of powerful tools is merely a warehouse; with it, even modest capabilities become productively deployable.

We propose, then, that *intelligence is knowing which function to apply.* This is not reductive in any deflationary sense. The space of functions is vast, and the problem of navigating it well is one of the deepest problems a cognitive system faces. But stating the claim clearly makes visible what had been implicit: that intelligence is fundamentally relational, holding between a system, a domain, and a repertoire of transformations, and that the system's task is always one of mapping.

3. The Universal Mathematical System

If intelligence is function selection, the space in which that selection occurs requires characterization. Here the foundational work of the twentieth century provides a decisive clarification.

The Gödelian and Church–Turing results together establish that the space of formal functions is, in a precise sense, fixed. Gödel's incompleteness theorems (Gödel, 1931) showed that any consistent formal system expressive enough to model arithmetic contains truths unprovable from within it—but crucially, those truths are still formally characterizable; they do not escape the formal domain. Church's lambda calculus (Church, 1936) and Turing's computability theory (Turing, 1936) showed that the class of effectively computable functions is a well-defined and exhaustively characterizable set. The Church–Turing thesis proposes that this set coincides with what is intuitively computable—that the formal and the intuitive converge.

One might object that this account leaves no room for genuine mathematical discovery—that if the space is fixed, mathematicians merely navigate it rather than

extend it. But this objection misunderstands the scope of the claim. The space of formal functions is fixed in the sense that any consistent formal structure is already, potentially, a member of it; but it is not fixed in the sense that all its members have been identified or that the relations among them are known. The discovery of a new mathematical structure is not the creation of a new formal possibility—it is the identification and articulation of one that was always implicit.

The key implication is that any mathematical system found applicable to the physical world is, by that fact, incorporated into the known set. It does not remain alien; it becomes an extension. This universality is precisely what Wigner (1960) found puzzling in his celebrated discussion of the unreasonable effectiveness of mathematics in the natural sciences. But what Wigner took as miraculous follows naturally from the universality of the formal space: physical structure finds mathematical description because the formal space is, in its meta-form, universally descriptive.

The human cognitive origin of mathematics—well documented in the embodied cognition literature (Lakoff & Núñez, 2000)—constrains what we reach for first and what strikes us as natural, but it does not constrain what is formally possible. Mathematics began with geometry and the arithmetic of countable physical things because human mathematical intuition is grounded in embodied experience. This is a fact about the history of access, not about the structure of what is accessible. Any mathematical system sufficiently alien to human intuition as to be currently unreachable would, upon discovery and verified application, simply join the set. The history of mathematics is a history of progressive access to a space that, in its formal envelope, was always complete.

4. The Architecture of Functions: Constants, Self-Reference, and Full Expansion

Not all functions are of the same type. A distinction with significant implications for both the foundations of mathematics and the structure of intelligence is the distinction between functions that define constants and functions that are self-referential.

The first class—*constant-defining functions*—takes no meaningful prior function as input and outputs a fixed value or set of values. These are the axiomatic generators: they just *are*. They do not justify themselves; they do not derive from anything prior. Their apparent arbitrariness is not a defect but a structural necessity—every formal system requires some ungrounded grounding, some initial characterization of the domain that is simply stipulated. Euclidean geometry requires its parallel postulate; Peano arithmetic requires its successor axiom; set theory requires its axioms

of extensionality and infinity. These constants make everything else possible.

The second class—*self-referential functions*—makes explicit or implicit reference to its own output, either directly (recursive functions that call themselves) or indirectly (systems of mutually defining functions). These functions appear, at first glance, self-sustaining. But this appearance is misleading. A recursive function is well-defined only when it has a base case—a point at which the self-reference terminates in a value that does not itself require further self-reference. That terminal value is always an instance of the first class: a constant.

When a self-referential function is fully expanded—when every recursive call is traced to its eventual ground—what remains is always a combination of transformations applied ultimately to constants. This is not merely a formal observation; it reflects the deeper point that Gödel's theorems illuminate. Any system capable of self-reference is also incapable of self-justification. The constants that ground the self-referential elaboration cannot themselves be derived from within the system.

A further observation: when fully expanded, the inputs to functions reveal themselves as, in many cases, other functions of the constant-defining type. That is, the base elements that a function ultimately operates on are not merely data-values but *generators*—procedures for producing the values that serve as inputs. The structure of intelligence, as a function-selection system, is thus grounded not merely in facts but in the generative procedures that produce facts.

5. Function Applicability: Shape, Resonance, and Fuzzy Emulation

The account of intelligence as function selection immediately generates a recognition problem: how does a system determine that a given function f is applicable to given data d? The answer cannot be that the system exhaustively checks all functions—the space is far too large. Nor can it be that the system applies a rule matching formal types—this merely relocates the problem to the meta-level of type-recognition.

We propose that applicability recognition operates through *structural resonance*: the data has a shape—a set of invariants—that matches, approximately and probabilistically, the expected input structure of the function. A function is recognized as applicable not through formal type-checking but through a process we call *fuzzy emulation*: a rapid, approximate simulation of f applied to a representation of the data, producing a preliminary output whose downstream connectivity can be assessed.

The relevant notion of shape is richer than mere syntactic type. It includes topological properties (the dimensionality and connectivity of the data space), symmetry properties (what transformations leave the data invariant), statistical properties

(the distribution, correlations, and moments of the data), causal properties (the directed structure of dependencies), and relational properties (the graph of relationships among data elements). A function is applicable when its expected input shape resonates sufficiently with the actual shape of the data for productive transformation to occur.

This fuzzy emulation mechanism is how experts in any domain appear to recognize relevant structure before fully engaging with it. The mathematician who looks at a problem and thinks "this feels like an induction argument" is running an approximate emulation of the induction schema against the problem's structure, before the proof has been found or verified. The physicist who reaches for group-theoretic methods upon seeing a certain equation is doing the same. What distinguishes expert from novice is not merely the size of the function repertoire but the calibration of the shape-matching prior—the learned mapping from structural features to applicable functions.

Crucially, the criterion of applicability is not purely structural but *compositional*. A function is deemed applicable here not merely because it can operate on this data but because its output will have a shape that serves as valid input to further functions. Applicability is evaluated in context—specifically, in the context of what the output needs to become. This introduces directionality into the account: the chain of function applications has an orientation, determined by what the downstream functions require.

6. Grounded Functions: Where Mathematics Meets World

Pure mathematical functions are, in a precise sense, atemporal. $f(x) = x^2$ does not happen at a time; it simply holds. The relationship between input and output is static, necessary, and not causal in any physical sense. This atemporality is what gives mathematics its generality and its independence from particular physical conditions.

But a distinctive class of functions exists that does not share this property. We call them *grounded functions*. A grounded function is one whose execution constitutes a causal event in the physical world: it is time-indexed, irreversible, and stochastic in ways that pure mathematical functions are not. The canonical examples are action()—any function whose execution changes the state of the world—and observe()—any function whose execution brings a sample of world-state into the processing system.

These functions are not purely mathematical in the standard sense, yet they are not outside the mathematical framework either. They are better understood as func-

tions at the interface between the formal system and its physical substrate—the points where abstract transformation makes causal contact with reality. The symbol grounding problem (Harnad, 1990)—the question of how formal symbols acquire genuine reference—finds its resolution precisely here: grounded functions are the mechanism by which the formal system is anchored to the world it describes.

The grounded functions serve two structural roles. First, they are the typical terminating functions of the applicability chain: the chain of mathematical transformations ends, at some point, not in a mathematical constant but in an action applied to the world. The applicability criterion—that output serves as input to further functions—bottoms out, eventually, in the grounded class. Second, they are the source of novelty in the system: because the world is not fully predictable, action() and observe() introduce information that was not present in the prior state of the formal system.

The observe() function deserves particular attention because of a property unique within the grounded class: it is the mechanism by which the axiomatic constants can be revised. The arbitrary foundational generators described in §4—the constants that underlie all self-referential elaboration—are not permanently fixed for a system embedded in the world. Through observe(), the system can acquire evidence that bears on its axiomatic commitments, and through iterated cycles of action() and observe(), the formal base of the system can evolve. This makes observe() the mechanism of learning in the deepest sense: not the updating of beliefs within a fixed framework but the potential revision of the framework itself.

The loop defined by action(), observe(), and the intervening chain of formal function applications is the fundamental architecture of embedded intelligent behavior—a perception-action cycle in the terminology of embodied cognition (Varela et al., 1991), a control loop in the language of cybernetics (Wiener, 1948), and a reinforcement learning environment in the terminology of contemporary machine learning (Sutton & Barto, 2018). The convergence of these descriptions across disciplines points toward a genuine structural insight: intelligent systems embedded in the world are all, at some level of abstraction, running this loop.

7. The Inertia of the Loop: On Bothering

We now arrive at the central question. The loop—action(), observe(), formal transformation, action()—is a powerful architecture. But architecture is not motive. A loop requires something to drive it; a chain of function applications requires something to orient it. What provides this?

The question can be sharpened. The applicability criterion—that output serves as input to further functions—is compositional, as noted in §5. But it is also regressive: if every function's applicability depends on its output being useful for further functions, and every further function's applicability depends on the same criterion, then either the chain regresses infinitely or it terminates in something. What does it terminate in?

We propose that, for biological systems, the chain terminates in *self-preservation*—and that the self-preservation drive arises from, and is indeed constituted by, a particular moment: the moment of self-recognition, when a system first develops a model that represents itself as a persistent entity in the world.

The moment deserves description. A biological system processes information long before it has a self-model. A bacterium chemotaxes toward glucose; a nematode navigates its environment; a fish avoids predators. All of this involves what we might call *functional teleology*—behavior organized around outcomes—without any requirement for self-representation. The inertia is there; what is not yet there is reflexivity.

At some point in cognitive development—and at some evolutionary point in the history of sufficiently complex nervous systems—a representational capacity emerges that is turned back on the system itself. The system develops a model not merely of the external world but of itself as an entity within that world: a persistent, bounded entity with survival conditions. This is the moment Metzinger (2003) identifies as the emergence of a phenomenal self-model—the content of a transparent self-representation that generates the first-person perspective.

What this moment creates, we propose, is not a new drive but the reflexive awareness of one that was already implicit. The system that previously avoided damage without knowing it was doing so now knows that it is an entity that can be damaged and can persist. That knowledge transforms the functional teleology into something more: a *stake*. The system now has an interest, in the philosophically relevant sense, in its own continuation. Every subsequent chain of function applications is, at the deepest level, an answer to the question: what should I do to continue?

This is why we bother. The drive requires no metaphysical addition to the physical picture. It is the ordinary consequence of a representational system complex enough to model itself and, having done so, finding that its own continuation is the most fundamental input constraint on everything it does. Spinoza (1677/1994) captured the underlying phenomenon in his doctrine of *conatus*—that every thing strives, insofar as it can, to persist in its own being. Our account provides a functional and representational specification of what that striving consists in and how it arises from

the structure of self-referential cognition.

8. The Cascade: From Self-Recognition to All Purpose

If self-preservation is the root inertia, how do we get from there to the full richness of human purpose—curiosity, love, aesthetic response, moral commitment, apparently selfless devotion to abstract causes?

The answer is that purpose cascades through abstraction layers, and that the further it travels from its root, the less recognizable the connection becomes. This is not a deficiency of the account but a feature: the cascade is precisely what allows complex motivational structure to arise from simple root conditions.

At the first level of abstraction, the cascade is straightforward. Hunger is self-preservation observed at the level of caloric state. Fear is self-preservation observed at the level of threat detection. Curiosity, already somewhat more removed, is self-preservation operating through uncertainty reduction: a system that cannot predict its environment is a system at risk, and so the drive to understand is continuous with the drive to survive, once sufficient cognitive complexity is present.

At higher levels of abstraction, the connections are harder to trace but structurally present. Social bonding is, in evolutionary terms, a self-preservation strategy for intensely social species—but once the bonding mechanisms are in place, they acquire motivational force independent of their origin, and love does not present itself as a survival strategy. Aesthetic response may originate in the fitness advantages of recognizing structure, pattern, and coherence in the environment—but in a sufficiently complex cognitive system, the response generalizes to abstract structure, and the mathematician's appreciation of a beautiful proof is continuous with, while not reducible to, the ancestral sensitivity to coherent form.

The abstraction layers are real elaborations, not disguised repetitions. When a person sacrifices their physical safety for a principle, the self-preservation root has been recursively applied to a self-model so abstract that the physical self is no longer the primary referent. The self that seeks preservation has itself been elaborated—the person identifies not only with their body but with their values, projects, relationships, and commitments. Preserving the self, at this level of abstraction, may require actions that damage the body. The root drive is intact; what has changed is the complexity of the self-model it operates on.

This account has a further implication. The purposes a system displays are sensitive to the content of its self-model. A system with a thin self-model—one representing only immediate physical state—will display only immediate physical purposes.

A system with a rich, abstract, temporally extended self-model will display purposes correspondingly rich, abstract, and extended. Intelligence and purpose are thus intertwined: richer intelligence generates richer self-models, which generate richer cascades of purpose.

9. The Incomplete Self and the Unending Loop

There is a formal complication to the account of §7 that deserves explicit treatment, because it illuminates both the structure of self-knowledge and the perpetual character of the observe() loop.

The self-model that wakes up and recognizes itself is a function running inside the system it models. This is a self-referential function in the sense described in §4—and it is therefore subject to the general limitation established there. Just as a recursive function cannot be self-sustaining without axiomatic base cases, and just as Gödel's theorems establish that no consistent formal system can fully characterize the truths of its own domain, a self-model operating within a cognitive system cannot fully represent that system.

This is not merely an empirical observation about the limitations of introspection, though it is consistent with those observations. It is a formal consequence of the structure of self-referential representation. The self-model is always, necessarily, an incomplete model of the self that generates it. There will always be aspects of the system's processing that fall outside the representational reach of the model—not because of insufficient effort, but because the completeness that would be required is formally impossible.

This Gödelian character of self-knowledge has several implications. It explains why the observe() loop never terminates—the system seeking to know itself fully will always find more to observe, because the self-model's domain of incompleteness is inexhaustible. It explains the unreliability of introspection as a structural consequence of the representational architecture, not merely as an empirical phenomenon. And it explains why self-knowledge has the phenomenological quality of inexhaustibility: the sense that there is always more to understand about oneself is not an illusion but an accurate perception of a formal fact (cf. Hofstadter, 1979).

The observe() loop, oriented by self-preservation and iterated through the action-observation cycle, is thus driven not only by external uncertainty but by internal incompleteness. Even in a world perfectly known, the self that acts in it would remain, to itself, a source of inexhaustible inquiry.

10. Open Questions: The Non-Biological Case

The framework developed in the preceding sections leaves one question conspicuously open, and we judge that the current state of evidence does not warrant a confident answer. The question is whether artificial intelligent systems—systems that optimize toward objectives but may lack phenomenal self-models—are running the same loop as biological systems, for the same reason.

There are two components to the question, separable in principle. The first is architectural: do artificial systems have the loop structure—action, observe, formal transformation, action—that we have identified as the fundamental pattern of embedded intelligence? The answer here is at least partially affirmative. Contemporary reinforcement learning systems have this structure explicitly; large language models have a version of it in the inference-and-response cycle. The architecture is present.

The second component is motivational: what drives the loop in artificial systems? Biological systems, on our account, are driven by self-preservation arising from a phenomenal self-model. Artificial systems have optimization targets—loss functions, reward signals—that provide direction without, as far as can currently be determined, arising from anything like self-recognition. The bacterium without a self-model has functional teleology; it is not clear that current artificial systems have more than this, despite their architectural sophistication.

The question of whether a sufficiently complex artificial system could develop something functionally equivalent to a self-model—and whether such a model would generate anything like the cascade of purpose described in §8—is not currently answerable from available data. We leave it, appropriately, for the next iteration of the observe() loop.

11. Conclusion: Why We Bother

We have argued that intelligence is, at its core, function selection; that the space of functions is formally universal and exhaustive; that function selection operates through structural resonance and fuzzy emulation; that grounded functions serve as the interface between the formal system and the physical world; and that the entire system has its inertia in self-preservation, arising from the moment a sufficiently complex system recognizes itself as a persistent entity with survival conditions.

The title question—why do we bother?—turns out to be the deepest question about intelligent systems, because it is prior to all questions about mechanism. We bother because, having woken up one day to find that we exist, we find that we have a stake. That stake is not derived from anything further; it is the bottom constant—the

ungrounded grounder—of the motivational system. From it, all purpose cascades.

The account is minimalist in its ontological commitments. It requires no vital force, no irreducible intentionality, no metaphysically special self. It requires only that representational systems of sufficient complexity will, when they model themselves, find that they are entities that can persist or fail to persist—and that this finding transforms functional organization into genuine motivation. The rest follows.

What does not follow, and what cannot follow from within the system, is a justification of the root. The self-preservation drive cannot justify itself; it can only be noted. It is the arbitrary constant that makes everything else determinate. We bother, in the end, not because bothering can be shown to be rationally necessary, but because we are systems that have found ourselves in a situation where the alternative is not to be.

That discovery, unrepeatable for each system that makes it, is the beginning of everything else.

References

Church, A. (1936). An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58(2), 345–363.

Damasio, A. (1994). *Descartes' Error: Emotion, Reason, and the Human Brain*. Putnam.

Gödel, K. (1931). Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38, 173–198.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1–3), 335–346.

Hofstadter, D. R. (1979). *Gödel, Escher, Bach: An Eternal Golden Braid*. Basic Books.

Kleene, S. C. (1936). General recursive functions of natural numbers. *Mathematische Annalen*, 112, 727–742.

Lakoff, G., & Núñez, R. E. (2000). *Where Mathematics Comes From: How the Embodied Mind Brings Mathematics into Being*. Basic Books.

Metzinger, T. (2003). *Being No One: The Self-Model Theory of Subjectivity*. MIT Press.

Spinoza, B. (1994). *Ethics* (E. Curley, Trans.). Princeton University Press. (Original work published 1677)

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2nd ed.). MIT Press.

Turing, A. M. (1936). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42(2), 230–265.

Varela, F. J., Thompson, E., & Rosch, E. (1991). *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press.

Wiener, N. (1948). *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press.

Wigner, E. P. (1960). The unreasonable effectiveness of mathematics in the natural sciences. *Communications on Pure and Applied Mathematics*, 13(1), 1–14.